An Introduction
to
BASIC COMPUTER CONCEPTS

CONTROL DATA
CORPORATION

An Introduction

to

BASIC COMPUTER CONCEPTS

Control Data Corporation
Minneapolis, Minnesota

Basic

DIGITAL COMPUTER CONCEPTS

Control Data Corporation
Minneapolis, Minnesota

TABLE OF CONTENTS

FOREWORD

A significant key to understanding future technical advances, as well as acquiring new knowledge, will be found in those concepts associated with electronic computers. Certainly, much of the responsibility - as well as the obligation - to educate people in these concepts, will fall to the schools. Thus, understanding computers - and their related concepts - is becoming increasingly more important and necessary to good education.

The installation of the electronic digital computer touches upon a critical area that is the concern of all educators - it involves change. As such, it requires learning new concepts, implementing new procedures, and evaluating advanced performance results.

In retrospect, beneficial change is welcomed by all educators; yet change which is suddenly initiated without preparation, and with little or no attention to understanding the problems involved, is unnecessarily difficult. The educator who has a basic understanding of some of the basic concepts of computers is able to accept and promote the transitional change harmoniously. It is toward this goal that this manual is hopefully dedicated.

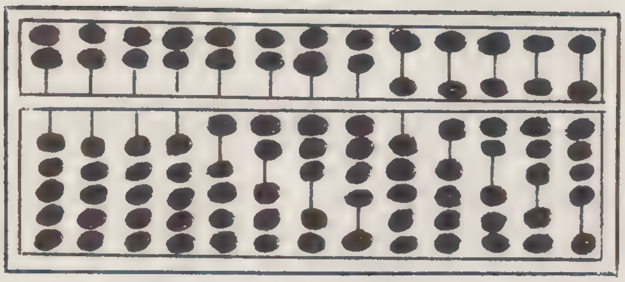
In the Beginning —

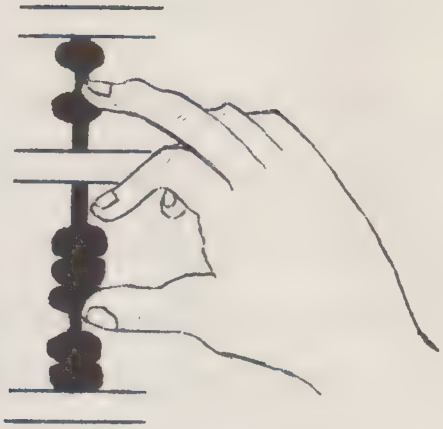
Great developments do not occur overnight! Behind the scenes - often hardly discernable - lie a long series of related and isolated events. Then like a great snowball, the day dawns when these past developments add up to a significant discovery.

The development of computers is somewhat analogous. Man has always liked to calculate - in fact, much of his environment has demanded calculation. His fingers became the initial adding machine. Later, the discovery and use of symbols advanced his capabilities to much greater horizons. Some men learned this symbolic language much better than others. They became experts - in fact, some established cults with well guarded "number secrets" which they refused to divulge to those outside their society.

About the same time, the growth of trade - on sea and land - required some medium of exchange and the money changers were born. It was but a short time until the money changers required advanced calculating techniques. Fortunately, the abacus made its appearance - a rather simple gadget with beads which could be moved easily up and down on wire strands. By placing different beads in different positions on the wire, the abacus merchant could calculate large amounts with surprising dexterity. Even at this time in history, there was some grumbling, and murmurs of discontent, from those merchants who could not acquire, or refused

to use, the abacus.





Even in those early days each neighborhood war seemed to bring out new calculators - or at least new techniques of calculation. War lords were quick to learn that brawn power pointed the catapult; whereas, brain power hit the target - and the artilleryman came into being. Today's missile is a direct descendent of this beginning!

And so it has continued through the pages of history; each step forward in trade, communication, industry - and warfare - has been either ushered into existence or followed by advances in calculations. And education has kept pace with this development.

For years, man dreamed of the possibility of constructing a

sophisticated machine which would perform the four basic steps of arithmetic: addition, subtraction, multiplication and division.

In 1812, Charles Babbage, the Rube Goldberg of his day, designed a machine which he called the "Difference Engine" which was to be an aid in the computation of mathematical tables. Later in 1833, he designed an "Analytical Engine" which was to be completely analytical. It was never built since the components it required could not be built with sufficient precision, at that period of history. Nevertheless, it had many of the elements of the modern day computer. The series of operations to solve a problem were to be punched on cards and fed into the machine as data. The memory was to hold 1000 words of 50 digits each - all in wheels. To modify a program, the scheme was to skip cards or move back to earlier cards.

In 1937, Professor Howard Aiken of Harvard University conceived a computer using the principles established by Babbage. This computer, The Automatic Sequence Controlled Calculator - known as Mark 1 - was completed in 1944 by the joint effort of Harvard and the International Business Machines Corporation. It could perform the four basic arithmetic operations and reference tables. Data was supplied by punched cards and the setting of manual switches. Answers were punched on cards or typed on a typewriter. A multiplication required 3 seconds. The Mark 1 was in use for over 15 years.

In 1946, J. P. Eckert and J. W. Mauchley at the University of Pennsylvania, designed the ENIAC (Electronic Numerical Integrator and Computer). It contained more than 18,000 vacuum tubes. Having no internal moving parts, it was a great advance over earlier machines.

Addition required 0.2 milliseconds, multiplication required 2.8 milliseconds, and numbers could be used up to 10 decimal digits. A program was established through plug-board wiring - a slow process which limited the use of the ENIAC.

In 1945, Dr. John von Neumann proposed a different type of electronic computer - an internally stored program. Internal storage, now used in all modern computers, offers greater versatility and flexibility to the use of computers.

The years immediately following World War II saw several significant events in computer development. At that time, a group of scientists, who formerly worked on a Navy research project decided to establish a computer development group in St. Paul, Minnesota. This group,which in 1946, became the nucleus of Electronic Research Associates - generally known as ERA - became a pioneer in digital computer development.

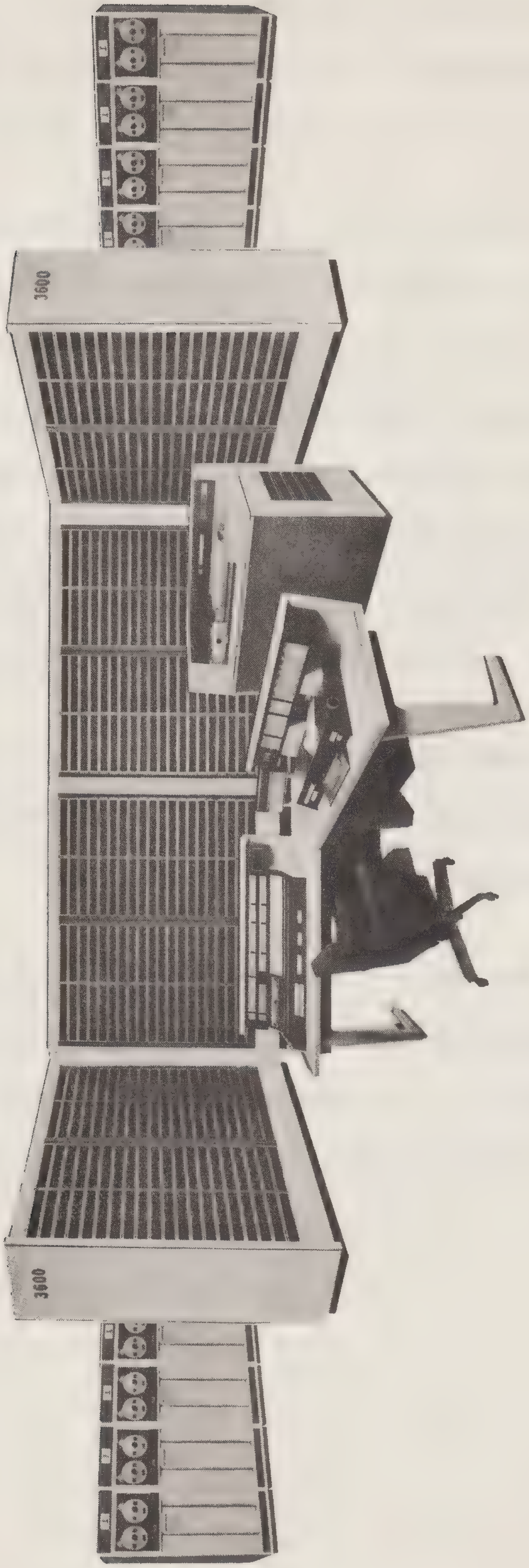
Of interest is the fact that W. C. Norris - now president of CDC - was Vice-President and General Manager of ERA until 1952 when ERA joined Remington Rand Univac. Mr. Norris served as Vice-

President of this operation until 1957 when RRU merged with Sperry Rand Incorporated. In 1957, Mr. Norris and a group of engineers formed the present Control Data Corporation.

Behind the scenes of this over simplified historical account lies a long road of complications; including the advance knowledge, the long years of research, and the step by step development of performance records, to that day when the electronic know-how made the computer possible. It's a road whose twists and turns are often indiscernable amidst the rush and turmoil of the modern world and it is easy to overlook the wide gulf between the day of the finger counter and the modern computer installation.

A Control Data Corporation engineer recently put this same thought into words when he spoke of his trip to Australia in regard to a computer installation.

"At first", he said, "I thought of the long way from home I had come to install a computer. But one day I saw a Bushman near the computer site. It was then that I realized I had traveled a distance that had to be measured in time rather than miles".


3600
3600

Why a Computer? —

A revelant preliminary question to a discussion of computers is - why are computers in demand in the first place? Actually, all reasons lead to the same answer - they are more economical! These economies are seldom apparent to the person who is not thoroughly acquainted with the details of the business or institution for whom the computer is installed. Nevertheless, the following factors are generally present and they provide additional evidence of the economies involved.

- Computers provide new time dimensions for the working day and for the human concept,
- Computers provide more efficient and effective controls over human error,
- Computers provide large capacities to store information - and the capability to rapidly access this information,
- Computers provide automatic organizational, administrative and management controls.

There are some operations, and some problems, where solutions, within practical time limitations, depend upon the speed of calculations. Consider, for example, the problem of multiplying 5000 pairs of numbers of 10 digits each. On a desk calculator, one such multiplication requires 10 seconds. Consequently, about 50,000 seconds, approximately 14 hours, is required for the 5000 multiplications. On a fast computer, <u>all</u> 5000 multiplications can be performed in less than 1/20 of <u>one</u> <u>second</u> - approximately one million times faster. Extending this analogy, it is not difficult to prove that many calculations would never be completed if man had to depend upon a desk calculator and his own speed of calculations.

In the matter of error control, computers make even more significant impact. The computer performs all operations - all calculations - internally, and without human intervention. Once a computer program has been written and checked out, the desired goals and objectives of the program will be attained unless the hardware (electronic and mechanical) components of the system malfunction. In contrast, a system of operations where human capabilities are significantly involved, is almost certain to contain many errors. This is particularly true where written records have to be continually monitored, edited and updated. In using a hand calculator or other common business machines, where manual entries are required, it is estimated that a person makes an error every 500 to 1000 operations. Computers, on the other hand, will run error-free for hours or days - and in one hour a large computer can perform more than a <u>billion</u> operations!

There are no limits to what the human memory can perform! Such statements are common, and there are many examples which indicate new and unexplored marvels of the mind - and the potential aspects of human memory. On the other hand, a practical and realistic analysis of human memory capabilities indicates significant limitations. In the first place, one must admit to <u>differences</u> - both in capacity and potential - in individuals. One also realizes that "remembering" is all too often a longer and more difficult process than that which is desired for efficient and effective business practice. Computers do have very fast access to tremendous amounts of stored

information; and this storage area is known as the computer "memory". In addition, computers have auxiliary storage available in the form of magnetic tapes, drums, disks, and other peripheral equipments. Literally, <u>millions</u> of facts - in coded format - can be stored in these devices. Of equal significance, is the fact that information stored in the computer system (in memory or auxiliary storage) can be <u>randomly</u> and <u>quickly</u> accessed by the computer. The access time is a matter of milliseconds (1 millisecond = 1/1000 of a second), microseconds (1 microsecond = 1/1,000,000 of a second), or nanoseconds (1 nanosecond = 1/1,000,000,000).

Finally, computers provide automatic control features which add strength to the structure of the business organization, which in turn, provides more security and returns to the whole enterprise. In recent years, "automation" has sometimes acquired an unenviable reputation as the scapegoat for unemployment. It is interesting to note that critics of automation voice and write their objections through publication and communication media that are made possible by modern automation; ride to work on the wheels of automation; reach their offices via automated elevators; buy foods which are processed and packaged by automation; wear clothes which are automatically tailored; and schedule all activities by time which is set by an automatic clock! Behind this criticism lives a world of progress and a society that is only possible with automation. The truth of the matter is that modern business depends upon automation in order to survive and prosper. Automation becomes the hope and helpmate of all workers!

If there is room for wonderment in contemplating the computer, much of it can be reserved for the internal speed of computers. In the past fifty years, man has made fantastic progress in his ability to cover distance in decreasing periods of time. The modern jet now travels 500 to 600 miles per hour. Satellites, on the other hand, travel across space at speeds up to 20,000 miles per hour. Yet the speed of computers is even more fantastic - even though it is a different type of speed. Computer speed is associated with executing instructions - that is, accomplishing calculations - rather than covering distances. Many of these tasks are arithmetical in nature, involving addition, subtraction, multiplication and division. Modern computers can execute an instruction in less than ten microseconds. (Control Data Corporation's 3100, 3200, 3600, and 6600 computers all have average instruction times of much less than ten microseconds.) A microsecond is one-millionth of one second. Thus, the modern computer can execute one million instructions in 10 seconds. It is difficult to comprehend the full significance of this tremendous speed. Some realization is gained if one attempts to estimate how long a time is required for the expert mathematician to add one million numbers of ten digits each; for example, numbers such as:

$$\begin{array}{r} 3254267 \\ \underline{2751649} \\ 6005916 \end{array}$$

Assuming each addition can be performed in five seconds, approximately 5,000,000 seconds are required to add one million numbers. Five million seconds is more than 173 eight hour days! Working eight hours per day for almost six months, the mathematician might possibly arrive at the same sum that the computer had calculated in 10 seconds!

Problems Conducive to Computers —

Some problems are more conducive to computer solution than others. Understanding this concept is helpful. Two questions are helpful in determining if a problem is conducive to computer solutions:

(1) Is it economically feasible?
(2) Is it quantitatively feasible?

Economically, the time and costs to prepare and program the problem, along with the cost of using the computer and its equipments must be considered. Some problems are either too simple, too small, or unimportant to justify the costs of computer solutions. However, it is worthy of note that problems which are considered unimportant and economically not feasible for computer solution under today's standards, may become much more attractive as computer techniques change.

Certainly some problems are simple enough, and used so sparingly, that it is not economically feasible to use computers. However, even a simple problem which must be solved for many individuals, or for many applications, may be better solved with computers. As an example, consider the income tax return which Joe Smith prepares each year for the Bureau of Internal Revenue. Normally, Joe spends a couple of hours making out his return and it appears to be uneconomical to prepare his return by computer application. On the

other hand, if Joe Smith's return can be programmed to fit the returns of thousands of other tax payers, a computer application appears in a more attractive light.

The computer, then, is particularly attractive in situations where one program solution can be used over and over, processing new problem parameters each time.

The iterative type problem is usually attractive for computer applications. This is a repetitious type problem which calculates successive results, so that each result comes closer to the true answer of the problem. This successive approximation procedure is commonly used to solve a certain class of mathematical problems. Where calculations of this nature are required, computers are very attractive, since the time to find thousands of approximations by other methods is impractical.

Problems situations which require constant and immediate measures of control are conducive to computer applications. For example, to control a process, an inventory, a buy and sell complex. In such areas, the amount of data that must be handled is considerable. Without computers, the control features of the system which include monitoring, decision making, editing and updating records, etc., are left completely to manual efforts. In some businesses, the amount of data involved in controlling current stock, providing storage, filling customer order, costing materials, and

adjusting for seasonal variations make it physically and economically impossible to operate without computers.

The second question - that of quantitative feasibility - is more insidious and less understood by most persons. All computers operate on numerical data. This means that alphabetic data must be converted to numerical codes before the computer can handle the data. This conversion is not a problem. More serious is the fact that some data, because of its nature, does not lend itself to numerical manipulation, and is much more difficult to program. For example, using a computer to advance a student's concepts in mathematics is much easier than trying to accomplish the same objective in English. Translating a paragraph of Russian language to English by a computer program is much more difficult than solving a differential equation. A useful rule to remember is this regard is:

> "That which infers capability of sight-vision is more difficult for the computer than that which infers capability to calculate".

It is interesting to note, that for humans, this rule is generally reversed. Thus, it is easier for a person to choose the two largest numbers from a group of 100 numbers than it is to find their sum. For the computer, the opposite is true. Of course, the computer will do both faster than the human!

CONTROL DATA 3200 COMPUTER SYSTEM

Three Computer Systems —

It is important to mark the growth and trend of computer utilization in three specific areas. Those briefly outlined here are:

- The Scientific System
- The Business System
- The Communication System

Ninety percent of <u>all</u> the world's scientists are alive today! Possibly 75% of these are working in, or performing work related to, Scientific Computing Systems. The Scientific System is distinguished by its devotion to analysis and calculation - either in specific areas or in research. Many of these facilities are part of governmental or industrial research laboratories, with a considerable number associated with colleges and universities. For most of these facilities, speed of calculation is the prime requisite, along with adequate internal storage capacity to hold results. Auxiliary storage, in many of these facilities is limited to magnetic tape units with the main emphasis upon holding translation programs (called software) such as, compilers, subroutines, and other calculation and language aids. Scientific groups were among the earliest to use computers. Working in areas where literally millions of calculations are involved, the computers have provided the speed and accuracy required to carry on the tremendous research and applications related to scientific endeavor.

The Business System and its use of computers is significantly different than the Scientific. Although the computer problems in both are similar in structure; the business problem is generally more concerned with input-output operations and operations on magnetic tape files. Large amounts of data are involved in most business applications. This data must be processed, recorded, filed, documented and updated. In addition, the business computer is concerned with effective and efficient techniques related to the business operation. Thus payroll problems, inventory, record keeping, purchasing, report preparations, sales data, billing, and numerous evaluation reports are involved. Most of these involve considerable input-output operations with limited calculation power.

In summary, computer business applications generally involve the following:

(a) Record information,
(b) Identify information,
(c) Check information for accuracy, redundancy,
(d) Format information for computer handling,
(e) Perform arithmetic computations on data,
(f) Sort and sequence data in some order,
(g) Update records with results of computations,
(h) Prepare output records or reports,
(i) Make statistical summaries.

How does the Computer Communications System differ from the Scientific or Business Processing Systems? Basically, the differences lie in how data is <u>received</u> and <u>delivered</u>. In the Communications System, data is brought into the computer center over tele-

phone, telegraph, data, or other communication lines. Computer servicing takes place on the data, and results are delivered to points of origin or other points via the same or different communication lines. The central computer must be tied into the input-output communication lines as well as other computers which may be involved in the whole complex. The fact that input-output lines must be serviced, at the exact moment in time, when data is available adds real time complexities to the programming that is required.

The communications concept has already had marked effects upon unique areas within the computer industry. These effects will no doubt expand in the years immediately before us.

Communication computer design requires that considerable more attention and detail be devoted to the programming characteristics of the communications system. The days of "black box" construction, without due regard to programming needs, are numbered. The area of error checking is a good example. Does the computer provide for hardware parity checking, or is this a programming function? Direct buffering of input-output data is a similar problem. Is this primarily a hardware or software function? Still another area is that of data validation. How much hardware should be provided to enable the programmer to validate data as it comes off the communication lines?

Finally, it is necessary to provide extensive input-output capabilities for the central processor to enable "cross talk" between

the various peripheral equipments, the input-output units, and other processing computers which may be part of the system. To provide for these capabilities, computers require coordinated and integrated systems of internal interrupts or scanning logics so that real time constraints do not become excessive for the programmer to handle.

The communications concepts has had considerable effect, and will continue to change the whole area involving engineers and programmers. Under the "black box philosophy", the engineer has not had to be too cognizant of the programmer's functions. However, under the communications concept, the engineer must learn the programming concepts in order to intergrate the hardware into the system.

On the other hand, the programmer of a communications system program is also forced to learn and become thoroughly acquainted with many of the hardware concepts. The programmer can no longer "leave all hardware problems to the engineer". A great deal of the success of a communications system program depends upon the programmer being able to understand the hardware characteristics of the various system hardware components involved. Thus, the communications system programmer who queues up messages on a drum, disk, or tape had better know the hardware characteristics of these devices.

The effect, then, upon engineers and programmers is one which demands that each group overlay the other in experience and knowledge. The communications concept thus fosters a "meeting of minds" between engineers and programmers, and for the first time, the system concept becomes most important. As a consequence, the professional workers are forced to relinquish those concepts that are closely associated to "individual functions" in favor of "system functions" - those functions which enable the whole complex of hardware <u>and</u> software to become an effective and efficient operating system.

In summarizing the communications concept, it is important to note the increasing interest in applying computer techniques to the needs of individuals. This trend is made possible by recent advances in communications technology. Under this concept, the computer system is designed to satisfy distinct requirements of many <u>individuals</u> who will use the computer. Under this concept, computer services can be made available to the "man in the street" for such functions as income tax returns, family budgets, aptitude testing, medical diagnoses, etc. It is entirely within reasonable conjecture to expect to find thousands of individuals using computers for such services in the next five years. The chances are very high that such services will come about through communications media being tied into computers!

CONTROL DATA 3600 COMPUTER SYSTEM

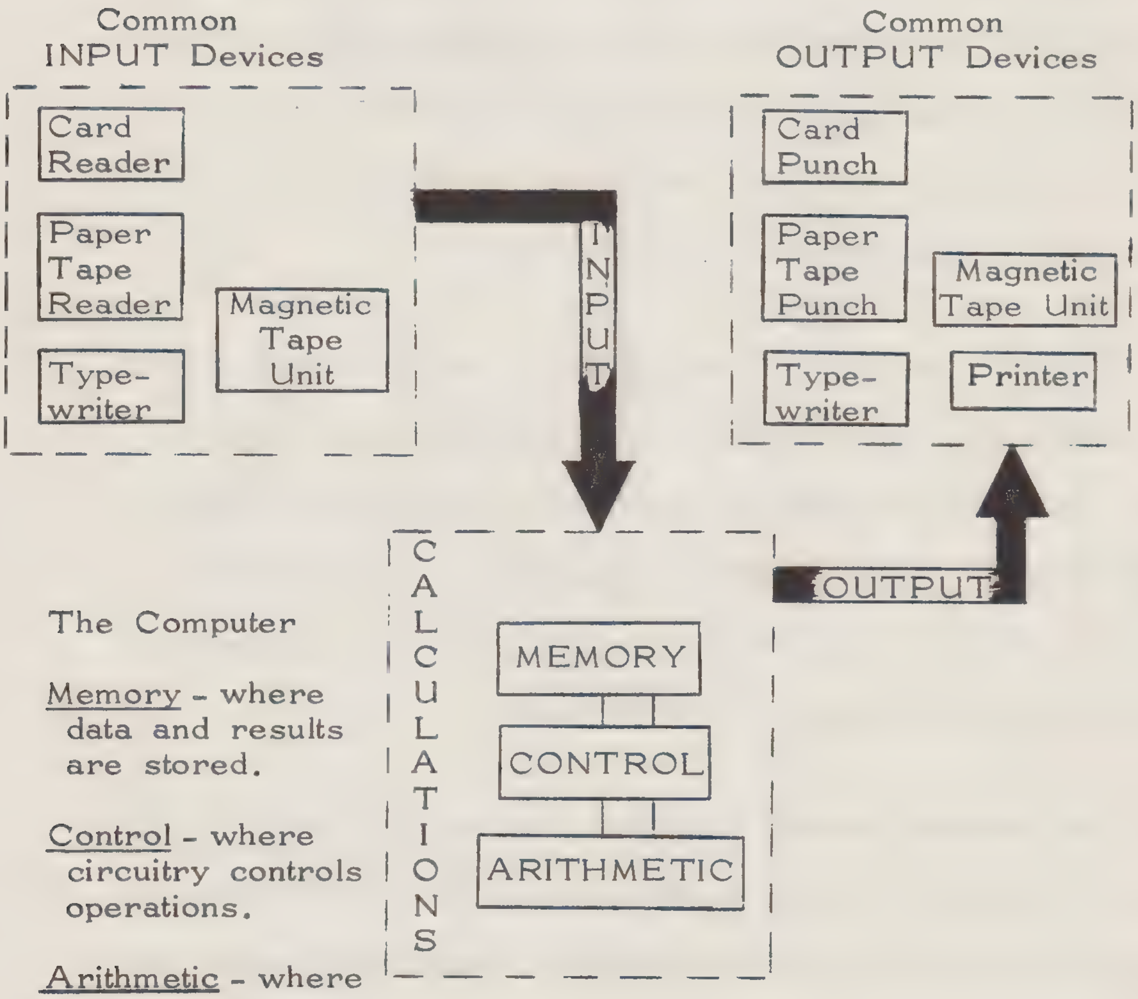
Basic Computer Configuration

Consider the ordinary procedures one follows to start his car each day. The ignition key is turned to a position (or a start button is depressed). This in turn provides battery power (INPUT) to turn the fly wheel, which causes a spark to ignite the combustible gas mixture and the motor starts. The proper gear setting (INSTRUCTION) engages the drive shaft and the car moves (OUTPUT). Other INSTRUCTIONS (applied in SEQUENCE) direct the car's speed and direction to its final destination. In summary, once provided an INPUT, the car motor responded to the input (ACTION or CALCULATIONS), and the car moved forward (OUTPUT).

Consider next, a decision to type a letter. One has a copy, either on scratch paper or in his mind - this is the INPUT. He types the desired characters (INSTRUCTIONS) in a predetermined SEQUENCE; the typewriter keys respond (ACTION or CALCULATION) and the printed characters appear on the letter page (OUTPUT).

The electronic digital computer, like the car, and the typewriter, consists of equipment designed and built for man to use. It has distinct and unique characteristics; along with its special purposes and functions. Whereas, the car is designed to pro-

vide point to point transportation; the computer is designed to perform calculation on numbers, make simple logic decisions, and control various other data handling equipments. Like the car, and the typewriter, one must provide INPUT - in the form of data; various calculations or ARITHMETIC are performed - on the data; and desired results are then made available (OUTPUT) - through some output device. The basic computer configurations can be simplified as follows:



The Computer

Memory - where data and results are stored.

Control - where circuitry controls operations.

Arithmetic - where all calculations take place.

The various INPUT and OUTPUT devices are chosen to fit the specific application. Many of these devices operate at much faster speeds than others. Consequently, speed of operations may be a criterion for choosing specific INPUT-OUTPUT equipments. Some have much larger capacities for storing data. For example, magnetic tapes can store much more data than cards or paper tape. It is important to note the following:

> All peripheral (INPUT-OUTPUT) devices operate at much <u>lower</u> speeds than the internal speed of the computer itself.

Whereas, computers operate in microseconds; the peripheral equipments operate in minutes, seconds, and at best - milliseconds. The reason for this is obvious, when one realizes that peripheral equipments must handle the hard copy (cards, paper, tape, type wheels, etc) of the system. Consequently, these devices must contain many more mechanical parts than the computer itself. In spite of this, the mechanical engineers have been able to advance the operating speeds of the peripherals to fantastic levels. Some of the more common operating speeds of these devices are given below:

Typewriter - INPUT - as fast as one can type.
OUTPUT - 10 characters per second.

Cards - INPUT - 1200 cards per minute.
OUTPUT - 100 cards per minute.

Paper Tapes - INPUT - 350 characters per second.
OUTPUT - 110 characters per second.

Magnetic Tapes - INPUT
OUTPUT 150-250 inches per second.

Printer - OUTPUT (only) - 1000 lines per minute
(each line = up to 128 characters)

The different storage capacities of these devices are also important and are given below:

Cards - 80 characters per card; where each column of the card can contain 1 character.

Paper Tapes - 10 characters per inch.

Magnetic Tapes - high density: 556 characters per inch.
low density: 200 characters per inch.

Printer - 120 to 128 characters per line.

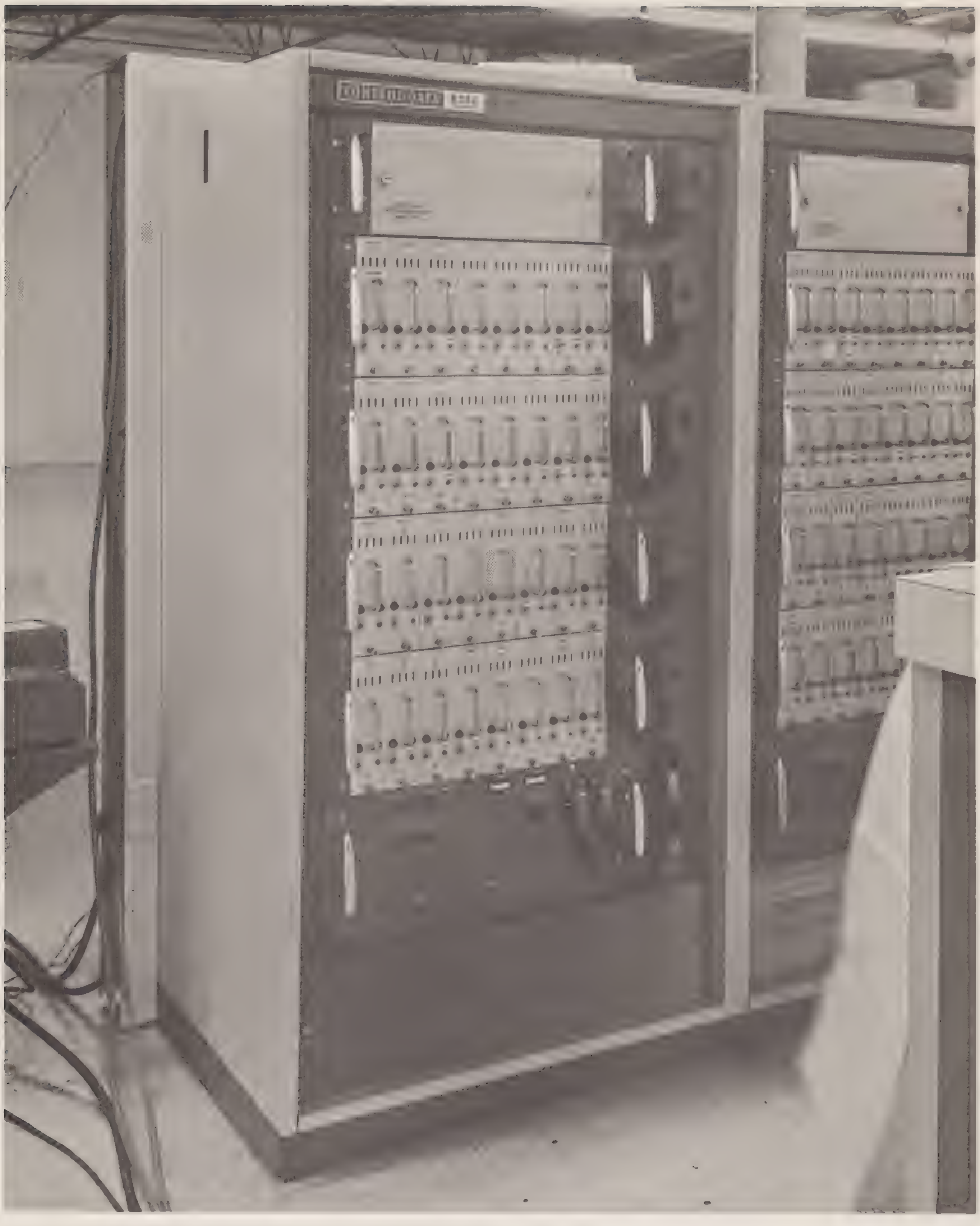
Because of the varying speeds and storage capacities, and the fact that computers are so much faster than the peripheral equipments, OFF LINE configurations are sometimes used. In this configuration, magnetic tapes are used as intermediate storage between punched cards and computer memory (on INPUT), and between computer memory and printer (on OUTPUT). Magnetic tapes are used since they are so much faster than card reading and paper printing. To

describe the OFF LINE procedure in more detail, the following example is given:

Assume a program which must take data from 12,000 cards. If the card reader is connected directly to the computer and the cards are read at 1200 cards per minute (ON LINE), it will take 10 minutes to read in the data. During these 10 minutes, the computer is simply waiting until all the cards are read in.

Instead of following this procedure, the 12,000 cards can be read by another card reader which is <u>not</u> connected to the computer, but which is connected to a Magnetic Tape Unit (OFF LINE). Again, there are 10 minutes involved, during which the data on the 12,000 cards is copied to the Magnetic Tape. During these 10 minutes, the computer can be processing and calculating another problem, since it is not connected to the off-line card reader.

The copied data - now on a Magnetic Tape - can be taken to the computer, placed on an ON LINE Magnetic Tape Unit and the computer can now read in the same data in approximately 1 to 6 <u>seconds</u> (since magnetic tapes can be read 100 to 600 times faster than cards).



Number System and Related Concepts

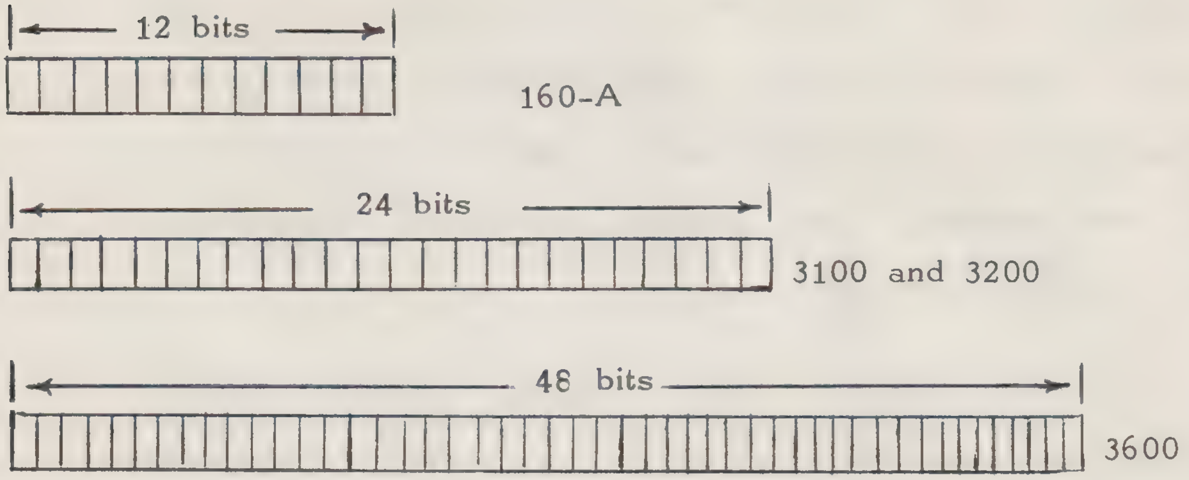
Computers use a numerical language in their internal operations. This numerical language is a two state system - known as <u>binary</u>. In the binary system, there are <u>two</u> possible digits: 0 and 1. Inside the computer these two states, 0 and 1, can be represented by various techniques: vacuum tube on or off, magnetic spots polarized in one direction or another, voltage high or low, a switch opened or closed.

A natural question is, how can larger numbers be represented if only zeros and ones are possible? This is accomplished by grouping the binary stages together. For example, assume there are three vacuum tubes as shown below. Each of these three tubes can be OFF or ON; thus, there are <u>eight</u> possibilities, and digits 0 through 7 can be represented by these three tubes <u>as a group</u>.

Tubes	Binary	Octal	Tubes	Binary	Octal
○ ○ ○	= 0 0 0	= 0	● ○ ○	= 1 0 0	= 4
○ ○ ●	= 0 0 1	= 1	● ○ ●	= 1 0 1	= 5
○ ● ○	= 0 1 0	= 2	● ● ○	= 1 1 0	= 6
○ ● ●	= 0 1 1	= 3	● ● ●	= 1 1 1	= 7

By grouping more than three tubes together, larger numbers can be represented. These <u>groups</u> of binary digits - known as

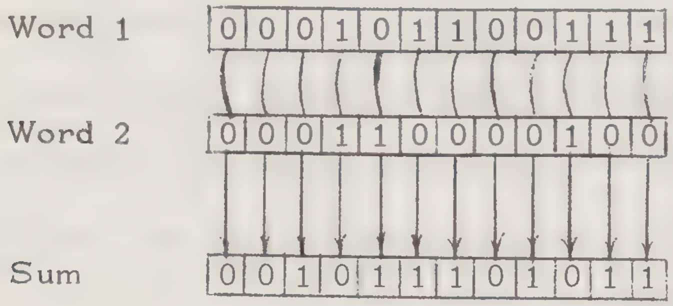
bits - determine the memory "word" size of various computers. For example, the 160-A Computer groups 12 of these bits in each internal word; whereas, the 3100 and 3200 group, 24 bits per word. The 3600 Computer, on the other hand, groups 48 bits.



The memory of a computer consists of thousands of these clusters or word groups. Each of the above computers, for example, contain 32,768 such words.

Of course, one can see that if vacuum tubes were used for each binary stage, that 32,768 times 48 tubes, (a total of 1,572,864) would be required for the large 3600. The heat problem alone, for so many tubes, would be prohibitive. Fortunately, transistors are used in larger computers. These take the place of tubes, are much smaller, and generate very little heat. On the other hand, transistors are more susceptible to heat and require more air conditioning.

One often hears the term "parallel arithmetic" in regard to computers. This term refers to the technique of performing the arithmetic on the corresponding bits of two words. Actually, arithmetic can take place at once - that is, all corresponding stages of the two words can be operated on at the same time. For example, assume two 12 bit words as follows:



Addition is performed all at once, each bit of one word being added to corresponding bit of the other word. If a "carry" results (1+1=10), the carry over (1) is added into the next bit.

All computers do not have "parallel" logic. Some perform arithmetic in a "serial" fashion. In this technique some bits are operated on, a partial result formed, then the next set of bits are operated on, etc. This is the same type of arithmetic that is normally taught in school. For example, to add 562 and 368, the following steps are used:

Add 2 and 8 = 10, carry 1, write ⟶ 0,
Add 6 and 6 + carry = 13, carry 1, write ⟶ 3,
Add 5 and 3 + carry = 9, write ⟶ 9,

Result = 930.

Parallel arithmetic is much faster than serial and is generally used in computers where high speed arithmetic is desired.

Another common computer term is "floating point arithmetic".

The opposite is "fixed point arithmetic". In fixed point arithmetic all numbers are treated as integers. For example, assume one wants to add 32.6 and 4.01 in a computer. If fixed point logic is used, the computer will add the two numbers as:

$$\begin{array}{r} 326 \\ \underline{401} \\ 727 \end{array}$$

Of course this is the wrong answer, and the question arises: How can a computer using fixed point arithmetic arrive at a correct result in such a situation? The solution, of course, is the responsibility of the programmer. In a situation such as this, he would have to do the above problem as follows:

$$\begin{array}{rr} \text{Multiply } 32.6 \text{ by } 100 = & 3260 \\ \text{Multiply } 4.01 \text{ by } 100 = & \underline{401} \\ \text{Sum} = & 3661 \end{array}$$

The programmer now knows that the result (3661) is 100 times too large and that the correct result is 36.61.

All of this previous procedure requires a great deal of adjustment by the programmer. In a complicated problem this bookkeeping (sometimes called "house-keeping") becomes very difficult. To eliminate this, "floating point arithmetic" is available. Using this technique, the exact location of the decimal point is known and established <u>by the computer</u> after each arithmetic operation. Although this appears to be an extraordinary capability, the technique is rather simple and is based upon the exponent - mantissa philos-

ophy common in logarithms.

Using the previous example: to add 32.6 and 4.01; 32.6 is stored as 02 3260, where:

02 = exponent of 10   3260 = fractional part of number assumed to be 0.3260.

Likewise, 4.01 is stored as 02 0401, where:

02 = exponent of 10   0401 = fractional part of number assumed to be 0.0401.

The addition gives:

```
02 3260
02 0401
-------
02 3661, which means:
```

10^2 times .3661 = 36.61, the correct answer.

In addition, the common exponents are not added.

It is important to note that floating point arithmetic can be built into a computer by circuitry (known as a hardware floating point package) or it can be programmed and stored in memory (software floating point package).

Control Data Corporation makes the hardware floating point package an optional offering on many of their computers. This permits various customers to choose the hardware or software approach in adding this feature for their specific application.

CONTROL DATA 160-A Computer System

The Stored Program Concept

Probably the most critical concept to understanding how a computer operates is the stored program concept. It is this feature, more than any other which distinguishes a computer from other machines. It is also this feature which sometimes lays a foundation of misunderstanding and dissatisfaction of what the computer does, and does not, accomplish.

Much of the difficulty of understanding this concept rises from the fact that most machines used by man are not stored program machines. The programs generally reside in man, and external to the machine. This is not true with the computer. In the computer, man literally relinguishes his control by storing a pre-determined program in the computer memory. This program then controls what the computer does.

Perhaps a useful analogy can be made by stretching the factors involved in operating the family car. Assume a special family car - that belonging to George Brown. George lived in Africa for years and when he returned to the United States he brought a chimpanzee home with him. The chimp was called "KULA" and was very intelligent. George had spent several years training KULA to drive the family car. He had also succeeded in teaching KULA to read and carry out the following five instructions.

START	KULA was able to start the family car when he saw this word.
LEFT	KULA would drive the car out the driveway and turn left.
RIGHT	KULA would drive the car out the driveway and turn right.
N-BLOCKS	KULA would drive "N" blocks; where N could be 1, 2, 3, - - 9.
HOME	KULA would drive the car back to the family garage.

On certain occasions, George was able to send the family car up or down the street by giving KULA these directions. When these situations arose, George really passed his control to KULA, and KULA became a stored program in the car. Once the program was written, and delivered to KULA, events could only occur by KULA's built in, habit forming capabilities.

Actually, man has always jealously guarded "control over events" as a sole responsibility. However, in the stored program of the computer, he relinguishes this control to the computer program. Understanding this concept, brings to the forefront, the importance of the stored program - and the individuals responsible for writing it. It also emphasizes the importance of cooperative analysis between the programmers and the customer who is installing the computer system. Only through careful preparation of the stored program, following detailed analysis by programmers and customer experts, is it possible to effectively and efficiently fulfill the objectives which led to the acquisition of the computer system.

Since the stored program is so important, a great deal of time must be spent on the analysis phase of the programming before any programming takes place. Of course, it is possible to write short sub-programs - or subroutines - which will be needed by the main program. However, the main program itself should not be programmed until complete analyses and agreements between programmers and customer officials have been completed. Long before the computer is delivered these analyses begin, and this helps explain the reason for the apparent "delay" that occurs between when the order is placed and the time when the computer system is installed. Actually, during this time, some of the most important work is going on - decisions will be made, which will affect the whole enterprise.

It is this area of computers which often puzzles those who do not understand the stored program concept. Americans are accustomed to buying something and "hopefully" using it immediately. Thus, the car buyer expects to drive his car away after signing the purchase order. This is not the true state of affairs with purchasing a computer. Because of the stored program concept, the immediate delivery of a computer would accomplish nothing except to display the equipment.

One might naturally ask the question: why not build the program circuitry into the computer as part of the hardware itself? This can be done, and if it is, the computer becomes a Special Purpose Computer. However, such a computer contains a most

serious disadvantage - it cannot be modified to meet changing conditions. Thus, it is completely inflexible. If the particular application is one in which <u>no</u> changes will ever occur, the Special Purpose Computer is possibly a reasonable alternative. In most situations, change is to be expected!

On the other hand, the <u>General Purpose Computer</u>, which depends upon a stored program for each problem situation, is most attractive to dynamic business or scientific applications. The stored program can be easily modified; portions of the program can be quickly deleted; and the program can be augmented by adding more instructions. Thus, as the business grows and as procedures change, the stored program can be modified to keep in step with these changes.

In summary then, the stored program requires careful analysis and coordination of efforts between programmers and customer officials. It requires time, and as a consequence, there is a period of time before delivery and installation during which the analyses are being made and the stored program is written. Once the program is written and checked out, however, the system not only provides error free solutions to the problem; it is also a rather simple and inexpensive matter to change the stored program to meet changing conditions. In the final analysis, one might say it is the stored program philosophy which has made computers versatile, flexible and sensitive to the pulse of changing requirements.

Basic Programming Concepts

The Computer Motel

A sixteen room motel of three floors has rooms numbered (addresses) as shown. The custodian, Archibald, is deaf and dumb, and his memory is a bit weak. If he is told to do more that one thing at a time, he will forget or confuse the instructions. Even with these limitations, he is quite efficient at accomplishing the things he can do. Archibald can only read and execute the five commands (instructions) which are shown on the next page. He is also able to find the rooms by the room numbers.

Rm 20	Rm 21	Rm 22	Rm 23	Rm 24	
Rm 15	Rm 16	Rm 17	Rm 18	Rm 19	
Rm 10	Rm 11	Rm 12	Rm 13	Rm 14	Office Rm 01

Letter boxes (registers) in each room hold the instructions for that room. Archy always travels from room to room in sequence unless he finds a GO TO instruction which breaks the sequence. After finishing, for example, the instruction in Box 22, he continues to Box 23 in Room 23, unless the instruction in Box 22 was a GO TO instruction.

The five instructions he has learned to read and execute are:

CLEAN	Clean the entire room
BED	Make the bed
FILL	Place new supplies in the room
GO TO	Go to the room number indicated instead of the next room in the sequence
STOP	Go back to the office - the program of work is done

The manager often finds it convenient to group the work by floors, and he likes to keep track of how Archibald is getting on with his work, so he has installed a letter box in the broom closet at the end of each floor (Rooms 14, 19, and 24). He frequently puts a GO TO 01 (go to the office) instruction in these boxes so that Archy will stop by the office to pick up an instruction in the custodian's box before starting the next job.

Archy has learned that after reading a card in the office, he is to throw this card in the waste basket. The office is the only room where he destroys a card. Archibald starts each day's work with the top card in the custodian's box in the office.

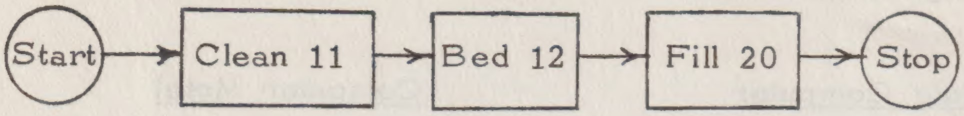
Tuesday night was rather quiet, so Wednesday morning, the manager walked around and made a list of the work to be done. He found that:

Room 11 needed cleaning.

Room 12 needed the bed made.

Room 20 needed supplies.

The manager made a flow chart, putting start and stop in circles and each single action in a rectangular box, as follows:



The manager then wrote up instruction cards and placed them as shown so that Archy could carry through the work in correct sequence. Thus, the manager is the programmer of this cleaning job.

The instruction, GO TO 11, is inserted in the box in the office

The instruction, CLEAN, is inserted in the box in Room 11

The instruction, BED, is inserted in the box of Room 12

The instruction, GO TO 20, is inserted in the box of Room 13

The instruction, FILL, is inserted in the box of Room 20

The instruction, STOP, is inserted in the box of Room 21

Does this sequence of instructions accomplish what the manager wanted done? If not, the manager, who is the programmer, made an error in the program.

A computer operates in a manner similar to the way Archy performs. First, a program is written, stored in the computer memory, the computer begins with the starting instruction, and progresses in sequence from instruction to instruction.

There are several analogies between an electronic computer and the preceding computer motel. Let us examine some of these by the following comparisons:

An Electronic Computer	Computer Motel
Transistors, diodes, wires, relays, fuses, framework, etc., make up the set and inflexible hardware traits which affect the speeds, functioning and types of programs.	The motel itself, the letter boxes, the set abilities of the custodian, etc., affect the types of programs.
Each computer contains a unique set or repertoire of instructions which can be executed. A repertoire of 50 to 74 instructions is built into the average computer.	The custodian could only understand and execute five instructions. This was his repertoire.
Instructions are executed, one at a time and in sequence, until a specific instruction causes the sequence to be changed.	The custodian performed one instruction at a time, visiting each room in sequence, unless he found a GO TO instruction.
Each instruction of a written computer program is stored in one register. Each register has a unique address associated with it. The total number of registers available for instructions is often referred to as, "computer memory" or "storage".	Each instruction of a written program is stored in a letter box. Each letter box has an address (the room number). The total number of letter boxes available (16) can be considered the motel memory storage.
Finally, the stored program is planned and created by man himself. The computer is a slave to the pre-stored strategy and creativity of the programmer. If logic of the programmer is false, it follows as night the day, the computer results will also be false. The machine is a sort of high speed moron. It does not think! It merely follows directions!	The status of custodian care at Computer Motel is entirely dependent upon the manager - who is the programmer. All the custodian can do is follow the instructions in each letter box. A dirty motel is the result of poor programming.